

AD-A101 851

CALIFORNIA UNIV BERKELEY OPERATIONS RESEARCH CENTER

F/6 12/1

MULTI-SERVER QUEUES. (U)

JUN 81 S M ROSS

AFOSR-81-0122

UNCLASSIFIED

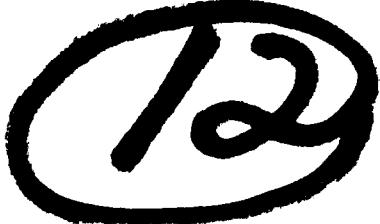
ORC-81-17

NL

(1)
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
10010
10011
10012
10013
10014
10015
10016
10017
10018
10019
10020
10021
10022
10023
10024
10025
10026
10027
10028
10029
10030
10031
10032
10033
10034
10035
10036
10037
10038
10039
10040
10041
10042
10043
10044
10045
10046
10047
10048
10049
10050
10051
10052
10053
10054
10055
10056
10057
10058
10059
10060
10061
10062
10063
10064
10065
10066
10067
10068
10069
10070
10071
10072
10073
10074
10075
10076
10077
10078
10079
10080
10081
10082
10083
10084
10085
10086
10087
10088
10089
10090
10091
10092
10093
10094
10095
10096
10097
10098
10099
100100
100101
100102
100103
100104
100105
100106
100107
100108
100109
100110
100111
100112
100113
100114
100115
100116
100117
100118
100119
100120
100121
100122
100123
100124
100125
100126
100127
100128
100129
100130
100131
100132
100133
100134
100135
100136
100137
100138
100139
100140
100141
100142
100143
100144
100145
100146
100147
100148
100149
100150
100151
100152
100153
100154
100155
100156
100157
100158
100159
100160
100161
100162
100163
100164
100165
100166
100167
100168
100169
100170
100171
100172
100173
100174
100175
100176
100177
100178
100179
100180
100181
100182
100183
100184
100185
100186
100187
100188
100189
100190
100191
100192
100193
100194
100195
100196
100197
100198
100199
100200
100201
100202
100203
100204
100205
100206
100207
100208
100209
100210
100211
100212
100213
100214
100215
100216
100217
100218
100219
100220
100221
100222
100223
100224
100225
100226
100227
100228
100229
100230
100231
100232
100233
100234
100235
100236
100237
100238
100239
100240
100241
100242
100243
100244
100245
100246
100247
100248
100249
100250
100251
100252
100253
100254
100255
100256
100257
100258
100259
100260
100261
100262
100263
100264
100265
100266
100267
100268
100269
100270
100271
100272
100273
100274
100275
100276
100277
100278
100279
100280
100281
100282
100283
100284
100285
100286
100287
100288
100289
100290
100291
100292
100293
100294
100295
100296
100297
100298
100299
100300
100301
100302
100303
100304
100305
100306
100307
100308
100309
100310
100311
100312
100313
100314
100315
100316
100317
100318
100319
100320
100321
100322
100323
100324
100325
100326
100327
100328
100329
100330
100331
100332
100333
100334
100335
100336
100337
100338
100339
100340
100341
100342
100343
100344
100345
100346
100347
100348
100349
100350
100351
100352
100353
100354
100355
100356
100357
100358
100359
100360
100361
100362
100363
100364
100365
100366
100367
100368
100369
100370
100371
100372
100373
100374
100375
100376
100377
100378
100379
100380
100381
100382
100383
100384
100385
100386
100387
100388
100389
100390
100391
100392
100393
100394
100395
100396
100397
100398
100399
100400
100401
100402
100403
100404
100405
100406
100407
100408
100409
100410
100411
100412
100413
100414
100415
100416
100417
100418
100419
100420
100421
100422
100423
100424
100425
100426
100427
100428
100429
100430
100431
100432
100433
100434
100435
100436
100437
100438
100439
100440
100441
100442
100443
100444
100445
100446
100447
100448
100449
100450
100451
100452
100453
100454
100455
100456
100457
100458
100459
100460
100461
100462
100463
100464
100465
100466
100467
100468
100469
100470
100471
100472
100473
100474
100475
100476
100477
100478
100479
100480
100481
100482
100483
100484
100485
100486
100487
100488
100489
100490
100491
100492
100493
100494
100495
100496
100497
100498
100499
100500
100501
100502
100503
100504
100505
100506
100507
100508
100509
100510
100511
100512
100513
100514
100515
100516
100517
100518
100519
100520
100521
100522
100523
100524
100525
100526
100527
100528
100529
100530
100531
100532
100533
100534
100535
100536
100537
100538
100539
100540
100541
100542
100543
100544
100545
100546
100547
100548
100549
100550
100551
100552
100553
100554
100555
100556
100557
100558
100559
100560
100561
100562
100563
100564
100565
100566
100567
100568
100569
100570
100571
100572
100573
100574
100575
100576
100577
100578
100579
100580
100581
100582
100583
100584
100585
100586
100587
100588
100589
100590
100591
100592
100593
100594
100595
100596
100597
100598
100599
100600
100601
100602
100603
100604
100605
100606
100607
100608
100609
100610
100611
100612
100613
100614
100615
100616
100617
100618
100619
100620
100621
100622
100623
100624
100625
100626
100627
100628
100629
100630
100631
100632
100633
100634
100635
100636
100637
100638
100639
100640
100641
100642
100643
100644
100645
100646
100647
100648
100649
100650
100651
100652
100653
100654
100655
100656
100657
100658
100659
100660
100661
100662
100663
100664
100665
100666
100667
100668
100669
100670
100671
100672
100673
100674
100675
100676
100677
100678
100679
100680
100681
100682
100683
100684
100685
100686
100687
100688
100689
100690
100691
100692
100693
100694
100695
100696
100697
100698
100699
100700
100701
100702
100703
100704
100705
100706
100707
100708
100709
100710
100711
100712
100713
100714
100715
100716
100717
100718
100719
100720
100721
100722
100723
100724
100725
100726
100727
100728
100729
100730
100731
100732
100733
100734
100735
100736
100737
100738
100739
100740
100741
100742
100743
100744
100745
100746
100747
100748
100749
100750
100751
100752
100753
100754
100755
100756
100757
100758
100759
100760
100761
100762
100763
100764
100765
100766
100767
100768
100769
100770
100771
100772
100773
100774
100775
100776
100777
100778
100779
100780
100781
100782
100783
100784
100785
100786
100787
100788
100789
100790
100791
100792
100793
100794
100795
100796
100797
100798
100799
100800
100801
100802
100803
100804
100805
100806
100807
100808
100809
100810
100811
100812
100813
100814
100815
100816
100817
100818
100819
100820
100821
100822
100823
100824
100825
100826
100827
100828
100829
100830
100831
100832<br

ORC 81-17
JUNE 1981

LEVEL II



MULTI-SERVER QUEUES

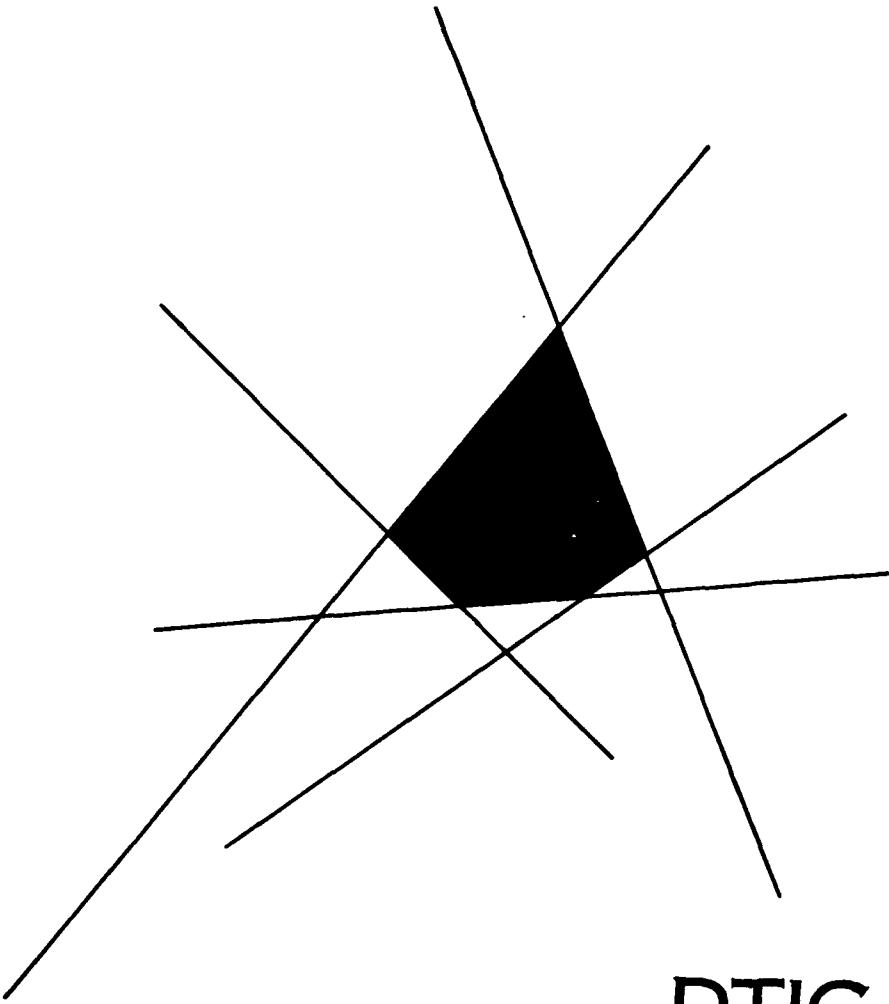
by

SHELDON M. ROSS

ADA101851

516 FILE COPY

OPERATIONS
RESEARCH
CENTER



DTIC
ELECTED
S JUL 23 1981 D
D

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

UNIVERSITY OF CALIFORNIA · BERKELEY

81 7 22 054

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/ _____	
Availability Codes _____	
Dist	Avail and/or Special
A	

MULTI-SERVER QUEUES

by

Sheldon M. Ross
 Department of Industrial Engineering
 and Operations Research
 University of California, Berkeley

DTIC
 ELECTE
 S JUL 23 1981 D
 D

JUNE 1981

ORC 81-17

This research was supported by the Air Force Office of Scientific Research (AFSC), USAF, under Grant AFOSR-81-0122 with the University of California. Reproduction in whole or in part is permitted for any purpose of the United States Government.

DISTRIBUTION STATEMENT A
Approved for public release; Distribution Unlimited

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER <i>(14) ORC-81-17</i>	2. GOVT ACCESSION NO. <i>AD-A202 852</i>	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) <i>(4) MULTI-SERVER QUEUES</i>	5. TYPE OF REPORT & PERIOD COVERED <i>(9) Research Report</i>	
6. PERFORMING ORG. REPORT NUMBER		
7. AUTHOR(s) <i>(10) Sheldon M. Ross</i>	8. CONTRACT OR GRANT NUMBER(s) <i>(13) AFOSR-81-0122✓</i>	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Operations Research Center University of California Berkeley, California 94720	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS <i>(11) 2304/A5 (11)</i>	
11. CONTROLLING OFFICE NAME AND ADDRESS United States Air Force Air Force Office of Scientific Research Bolling Air Force Base, D.C. 20332	12. REPORT DATE <i>(11) June 1981</i>	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. NUMBER OF PAGES <i>(11) 16</i>	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.	15. SECURITY CLASS. (of this report) Unclassified	
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Multiserver Erlang Loss Processor Sharing Finite Capacity M/G/k G/M/k	Reversed Process	
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) (SEE ABSTRACT)		

MULTI-SERVER QUEUES

Sheldon M. Ross

Department of Industrial Engineering and
Operations Research

ABSTRACT

We will survey a variety of multiserver models in which the arrival stream is a Poisson process. In particular, we will consider the Erlang loss model in which arrivals finding all servers busy are lost. In this system, we assume a general service distribution. We will also consider finite and infinite capacity versions of this model. Another model of this type is the shared processor system in which service is shared by all customers.

Another model to be considered is the G/M/k in which arrivals are in accordance with a renewal process and the service distribution is exponential. We will analyze this model by means of the embedded Markov chain approach.

0. INTRODUCTION

We will consider some multiserver queueing models. In Section 1, we deal with the Erlang loss model which supposes Poisson arrivals and a general service distribution G . By use of a "reversed process" argument (see [2]) we will indicate a proof of the well-known result that the distribution of number of busy servers depends on G only through its mean. In Section 2 we then analyze a shared-processor model in which the servers are able to combine forces. Again making use of the reverse process, we obtain the limiting distribution for this model. In Section 3 we review the embedded Markov chain approach for the G/M/k model; and in the final section we present the model M/G/k.

1. THE ERLANG LOSS SYSTEM

One of the most basic types of queueing system are the loss systems in which an arrival that finds all servers busy is presumed lost to the system. The simplest such system is the M/M/k loss system in which customers arrive according to a Poisson process having rate λ , enter the system if at least one of the k servers is free, and then spend an exponential amount of time with rate μ being served. The balance equations for the stationary probabilities are

$$\begin{array}{ll} \text{State} & \text{Rate leave = rate enter} \\ 0 & \lambda P_0 = \mu P_1 \\ i, 0 < i < k & (\lambda + i\mu)P_i = (i+1)\mu P_{i+1} + \lambda P_{i-1} \\ k & k\mu P_k = \lambda P_{k-1} . \end{array}$$

Using the equation $\sum_0^k P_i = 1$, the above equations can be solved to give

$$P_i = \frac{(\lambda/\mu)^i / i!}{\sum_{j=0}^k (\lambda/\mu)^j / j!}, \quad i = 0, 1, \dots, k.$$

Since $E[S] = 1/\mu$, where $E[S]$ is the mean service time, the above can be written as

$$P_i = \frac{(\lambda E[S])^i / i!}{\sum_{j=0}^k (\lambda E[S])^j / j!}, \quad i = 0, 1, \dots, k.$$

The above was originally obtained by Erlang who then conjectured that it was valid for an arbitrary service distribution. We shall present a proof of this result, known as the Erlang loss formula when the service distribution G is continuous and has density g .

Theorem 1. The limiting distribution of the number of customers in the Erlang loss system is given by

$$P\{n \text{ in system}\} = \frac{(\lambda E[S])^n / n!}{\sum_{i=0}^k (\lambda E[S])^i / i!}, \quad n = 0, 1, \dots, k$$

and given that there are n in the system, the ages (or the residual times) of these n are independent and identically distributed according to the equilibrium distribution of G .

Proof. We can analyze the above system as a Markov process by letting the state at any time be the ordered ages of the customers in service at that time. That is, the state will be $\underline{x} = (x_1, x_2, \dots, x_n)$, $x_1 \leq x_2 \leq \dots \leq x_n$, if there are n customers in service, the most recent one having arrived x_1 time units ago, the next most recent arrival being x_2 time units ago, and so on. The process of successive states will be a Markov process in the sense that the conditional distribution of any future state given the present and all the past states will depend only on the present state. In addition, let us denote by $\lambda(t) = g(t)/\bar{G}(t)$ the hazard rate function of the service distribution.

We will attempt to use the reverse process to obtain the limiting probability density $p(x_1, x_2, \dots, x_n)$, $1 \leq n \leq k$, $x_1 \leq x_2 \leq \dots \leq x_n$, and $P(\phi)$ the limiting probability that the system is empty. Now since the age of a customer in service increases linearly from 0 upon its arrival to its service time upon its departure, it is clear that if we look backwards, we will be following the excess or additional service time of a customer. As there will never be more than k in the system, we make the following conjecture.

Conjecture. In steady state, the reverse process is also a k server loss system with service distribution G in which arrivals occur according to a Poisson process with rate λ . The state at any time represents the ordered residual service times of customers presently in service. In addition, the limiting probability density is

$$p(x_1, \dots, x_n) = \frac{\lambda^n \prod_{i=1}^n \bar{G}(x_i)}{\sum_{i=0}^k (\lambda E[S])^i / i!}, \quad x_1 \leq x_2 \leq \dots \leq x_n$$

and

$$P(\phi) = \left[\sum_{i=0}^k (\lambda E[S])^i / i! \right]^{-1}.$$

To verify the conjecture, for any state $\underline{x} = (x_1, \dots, x_i, \dots, x_n)$, let $e_i(\underline{x}) = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$. Now in the original process when the state is \underline{x} , it will instantaneously go to $e_i(\underline{x})$ with a probability density equal to $\lambda(x_i)$ since the

person whose time in service is x_i would have to instantaneously complete its service. Similarly in the reversed process if the state is $e_i(\underline{x})$, then it will instantaneously go to \underline{x} if a customer having service time x_i instantaneously arrives. So we see that

in forward: $\underline{x} \rightarrow e_i(\underline{x})$ with probability intensity $\lambda(x_i)$
 in reverse: $e_i(\underline{x}) \rightarrow \underline{x}$ with (joint) probability intensity $\lambda g(x_i)$.

Hence if $p(\underline{x})$ represents the limiting density, then we would need that

$$p(\underline{x})\lambda(x_i) = p(e_i(\underline{x}))\lambda g(x_i)$$

or, since $\lambda(x_i) = g(x_i)/\bar{G}(x_i)$,

$$p(\underline{x}) = p(e_i(\underline{x}))\lambda\bar{G}(x_i)$$

which is easily seen to be satisfied by the conjectured $p(\underline{x})$.

To complete our proof of the conjecture, we must consider transitions of the forward process from \underline{x} to $(0, \underline{x}) = (0, x_1, x_2, \dots, x_n)$ when $n < k$. Now

in forward: $\underline{x} \rightarrow (0, \underline{x})$ with instantaneous intensity λ
 in reverse: $(0, \underline{x}) \rightarrow \underline{x}$ with probability 1.

Hence we must verify that

$$p(\underline{x})\lambda = p(0, \underline{x})$$

which easily follows since $\bar{G}(0) = 1$.

Hence we see that the conjecture is true and so, upon integration, we obtain

$$P\{n \text{ in the system}\}$$

$$= P(\phi)\lambda^n \int_{x_1 \leq x_2 \leq \dots \leq x_n} \prod_{i=1}^n \bar{G}(x_i) dx_1 dx_2 \dots dx_n$$

$$= P(\phi) \frac{\lambda^n}{n!} \int_{x_1, x_2, \dots, x_n} \prod_{i=1}^n \bar{G}(x_i) dx_1 dx_2 \dots dx_n$$

$$= P(\phi)(\lambda E[S])^n/n!, n = 1, 2, \dots, k$$

where $E[S] = \int \bar{G}(x)dx$ is the mean service time. Also, we see that the conditional distribution of the ordered ages given that there are n in the system is

$$p(\underline{x} | n \text{ in the system}) = p(\underline{x}) / P\{n \text{ in the system}\}$$

$$= n! \prod_{i=1}^n (\bar{G}(x_i)/E[S]).$$

As $\bar{G}(x)/E[S]$ is just the density of G_e , the equilibrium distribution of G , this completes the proof. ||

In addition, by looking at the reversed process, we also have the following corollary.

Corollary 1. In the Erlang loss model, the departure process (including both customers completing service and those that are lost) is a Poisson process at rate λ .

Proof. The above follows since in the reversed process arrivals of all customers (including those that are lost) constitutes a Poisson process.

2. THE SHARED PROCESSOR SYSTEM

Suppose that customers arrive in accordance with a Poisson process having rate λ . Each customer requires a random amount of work, distributed according to G . The server can process work at a rate of one unit of work per unit time, and divides his time equally among all of the customers presently in the system. That is, whenever there are n customers in the system, each will receive service work at a rate of $1/n$ per unit time.

Let $\lambda(t)$ denote the failure rate function of the service distribution, and suppose that $\lambda E[S] < 1$ where $E[S]$ is the mean of G .

To analyze the above, let the state at any time be the ordered vector of the amounts of work already performed on customers still in the system. That is, the state is $\underline{x} = (x_1, x_2, \dots, x_n)$, $x_1 \leq x_2 \leq \dots \leq x_n$ if there are n customers in the system and x_1, \dots, x_n is the amount of work performed on these n customers. Let $p(\underline{x})$ and $P(\phi)$ denote the limiting probability density and the limiting probability that the system is empty. We make the following conjecture regarding the reverse process.

Conjecture. The reverse process is a system of the same

type, with customers arriving at a Poisson rate λ , having workloads distributed according to G and with the state representing the ordered residual workloads of customers presently in the system.

To verify the above conjecture and at the same time obtain the limiting distribution let $e_i(\underline{x}) = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ when $\underline{x} = (x_1, \dots, x_n)$, $x_1 \leq x_2 \leq \dots \leq x_n$. Note that

in forward: $\underline{x} \rightarrow e_i(\underline{x})$ with probability intensity $\frac{\lambda(x_i)}{n}$
 in reverse: $e_i(\underline{x}) \rightarrow \underline{x}$ with (joint) probability intensity $\lambda G'(x_i)$.

The above follows as in the previous section with the exception that if there are n in the system then a customer who already had the amount of work x_i performed on it will instantaneously complete service with probability $\lambda(x_i)/n$.

Hence, if $p(\underline{x})$ is the limiting density then we need that

$$p(\underline{x}) \frac{\lambda(x_i)}{n} = p(e_i(\underline{x})) \lambda G'(x_i)$$

or, equivalently,

$$\begin{aligned} p(\underline{x}) &= n \bar{G}(x_i) p(e_i(\underline{x})) \lambda \\ &= n \bar{G}(x_i) (n - 1) \bar{G}(x_j) p(e_j(\underline{x})) \lambda^2, \quad i \neq j \\ &\vdots \\ &= n! \lambda^n P(\phi) \prod_{i=1}^n \bar{G}(x_i). \end{aligned} \tag{1}$$

Integrating over all vectors \underline{x} yields

$$P\{n \text{ in system}\} = (\lambda E[S])^n P(\phi).$$

Using

$$P(\phi) + \sum_{n=1}^{\infty} P\{n \text{ in the system}\} = 1$$

gives

$$P\{n \text{ in the system}\} = (\lambda E[S])^n (1 - \lambda E[S]), \quad n \geq 0.$$

Also, the conditional distribution of the ordered amounts of

work already performed, given n in the system is, from (1)

$$p(\underline{x} \mid n) = p(\underline{x}) / P\{n \text{ in system}\}$$

$$= n! \prod_{i=1}^n (\bar{G}(x_i) / E[S]) .$$

That is, given n customers in the system the unordered amount of work already performed are distributed independently according to G_e , the equilibrium distribution of G .

All of the above is based on the assumption that the conjecture is valid. To complete the proof of its validity, we must verify that

$$p(\underline{x})\lambda = p(0, \underline{x}) \frac{1}{n+1} .$$

The above being the relevant equation since the reverse process when in state $(\epsilon, \underline{x})$ will go to state \underline{x} in time $(n+1)\epsilon$. As the above is easily verified, we have thus shown

Theorem 2. For the Processor Sharing Model, the number of customers in the system has the distribution

$$P\{n \text{ in system}\} = (\lambda E[S])^n (1 - \lambda E[S]) , n \geq 0 .$$

Given n in the system, the completed (or residual) workloads are independent and have distribution G_e . The departure process is a Poisson process with rate λ .

If we let L denote the average number in the system, and W , the average time a customer spends in the system then

$$\begin{aligned} L &= \sum_{n=0}^{\infty} n(\lambda E[S])^n (1 - \lambda E[S]) \\ &= \frac{\lambda E[S]}{1 - \lambda E[S]} . \end{aligned}$$

We can obtain W from the well-known formula $L = \lambda W$ and so

$$W = L/\lambda = \frac{E[S]}{1 - \lambda E[S]} .$$

Another interesting computation in this model is that of the conditional mean time an arrival spends in the system given its workload is y . To compute this quantity, fix y and say that a customer is "special" if its workload is between y and

$y + \epsilon$. By $L = \lambda W$, we thus have that

$$\begin{aligned} & \text{Average Number of Special Customers in the System} \\ &= \text{Average Arrival Rate of Special Customer} \times \text{Average} \\ & \quad \text{Time a Special Customer Spends in the System.} \end{aligned}$$

To determine the average number of special customers in the system, let us first determine the density of the total workload of an arbitrary customer presently in the system. Suppose such a customer has already received the amount of work x . Then the conditional density of its workload is

$$f(w \mid \text{has received } x) = g(w)/\bar{G}(x), \quad x \leq w.$$

But, from Theorem 2, the amount of work an arbitrary customer in the system has already received has the distribution G_e . Hence the density of the total workload of someone present in the system is

$$\begin{aligned} f(w) &= \int_0^w \frac{g(x)}{\bar{G}(x)} dG_e(x) \\ &= \int_0^w \frac{g(x)}{E[S]} dx, \quad \text{since } dG_e(x) = \bar{G}(x)/E[S] \\ &= wg(w)/E[S]. \end{aligned}$$

Hence the average number of special customers in the system is

$$\begin{aligned} & E[\text{number in system having workload between } y \text{ and } y + \epsilon] \\ &= Lf(y)\epsilon + o(\epsilon) \\ &= Lyg(y)\epsilon/E[S] + o(\epsilon). \end{aligned}$$

In addition, the average arrival rate of customers whose workload is between y and $y + \epsilon$ is

$$\text{Average arrival rate} = \lambda g(y)\epsilon + o(\epsilon).$$

Hence we see that

$$\begin{aligned} & E[\text{time in system} \mid \text{workload in } (y, y + \epsilon)] \\ &= \frac{Lyg(y)\epsilon}{E[S]\lambda g(y)\epsilon} + \frac{o(\epsilon)}{\epsilon}. \end{aligned}$$

Letting $\varepsilon \rightarrow 0$, we obtain

$$\begin{aligned} E[\text{time in system} \mid \text{workload is } y] &= \frac{y}{\lambda E[S]} L \\ &= \frac{y}{1 - \lambda E[S]} . \end{aligned}$$

Thus the average time in the system of a customer needing y units of work also depends on the service distribution only through its mean.

3. THE G/M/k QUEUE

In this model we suppose that there are k servers, each of whom serves at an exponential rate μ . We allow the time between successive arrivals to have an arbitrary distribution G . In order to ensure that a steady-state (or limiting) distribution exists, we assume $1/\mu_G < k\mu$ where μ_G is the mean of G .

To analyze this model, we will use an embedded Markov chain approach. Define X_n as the number in the system as seen by the n th arrival. Then it is easy to see that $\{X_n, n \geq 0\}$ is a Markov chain.

To derive the transition probabilities of the Markov chain, it helps to note the relationship

$$X_{n+1} = X_n + 1 - Y_n, \quad n \geq 0$$

where Y_n denotes the number of departures during the interarrival time between the n th and $(n+1)$ st arrival. The transition probabilities can be calculated as

Case (i): $j > i + 1$. In this case, $P_{ij} = 0$.

Case (ii): $j \leq i + 1 \leq k$. In this case,

$P_{ij} = P\{i+1-j \text{ of } i+1 \text{ services are completed in an interarrival time}$

$$= \int_0^\infty P\{i+1-j \text{ of } i+1 \text{ are completed} \mid \text{interarrival time is } t\} dG(t)$$

$$= \int_0^\infty \binom{i+1}{j} (1 - e^{-\mu t})^{i+1-j} (e^{-\mu t})^j dG(t).$$

Case (iii): $i + 1 \geq j \geq k$. To evaluate P_{ij} , in this case, we first note that when all servers are busy the departure process is a Poisson process with rate $k\mu$. Hence,

$$\begin{aligned} P_{ij} &= \int_0^{\infty} P\{i+1-j \text{ departures in time } t\} dG(t) \\ &= \int_0^{\infty} e^{-k\mu t} \frac{(k\mu t)^{i+1-j}}{(i+1-j)!} dG(t). \end{aligned}$$

Case (iv): $i + 1 \geq k > j$. Conditioning first on the interarrival time and then on the time until there are only k in the system (call this latter random variable T_k) yields

$$\begin{aligned} P_{ij} &= \int_0^{\infty} P\{i+1-j \text{ departures in time } t\} dG(t) \\ &= \int_0^{\infty} \int_0^t P\{i+1-j \text{ departures in } t \mid T_k = s\} \\ &\quad k\mu e^{-k\mu s} \frac{(k\mu s)^{i-k}}{(i-k)!} ds dG(t) \\ &= \int_0^{\infty} \int_0^t \binom{k}{j} (1 - e^{-\mu(t-s)})^{k-j} (e^{-\mu(t-s)})^j \\ &\quad k\mu e^{-k\mu s} \frac{(k\mu s)^{i-k}}{(i-k)!} ds dG(t). \end{aligned}$$

We now can verify by a direct substitution into the equations $\pi_j = \sum_i \pi_i P_{ij}$ that the limiting probabilities of this Markov chain are of the form

$$\pi_{k-1+j} = c\beta^j, \quad j = 0, 1, \dots.$$

Substitution into any of the equations $\pi_j = \sum_i \pi_i P_{ij}$ when $j > k$ yields that β is given as the solution of

$$\beta = \int_0^{\infty} e^{-k\mu t(1-\beta)} dG(t) .$$

The values $\pi_0, \pi_1, \dots, \pi_{k-2}$, can be obtained by recursively solving the first $k - 1$ of the steady-state equations, and c can then be computed by using $\sum_0^x \pi_i = 1$.

If we let W_Q^* denote the amount of time that a customer spends in queue, then we can show, upon conditioning, that

$$W_Q^* = \begin{cases} 0 & \text{with probability } \sum_0^{k-1} \pi_i = 1 - \frac{c\beta}{1-\beta} \\ \text{Exp}(k\mu(1-\beta)) & \text{with probability } \sum_k^{\infty} \pi_i = \frac{c\beta}{1-\beta} \end{cases}$$

where $\text{Exp}(k\mu(1-\beta))$ is an exponential random variable with rate $k\mu(1-\beta)$.

4. THE FINITE CAPACITY M/G/k

In this section, we consider an M/G/k queuing model having finite capacity N . That is, a model in which customers, arriving in accordance with a Poisson process having rate λ , enter the system if there are less than N others present when they arrive, and are then serviced by one of k servers, each of whom has service distribution G . Upon entering, a customer will either immediately enter service if at least one server is free or else join the queue if all servers are busy.

Our objective is to obtain an approximation for W_Q , the average time an entering customer spends waiting in queue. To get started we will make use of the idea that if a (possibly fictitious) cost structure is imposed, so that entering customers are forced to pay money (according to some rule) to the system, then the following identity holds--namely,

time average rate at which the system earns = average arrival rate of entering customers \times average amount paid by an entering customer.

By choosing appropriate cost rules, many useful formulae can be obtained as special cases. For instance, by supposing

that each customer pays \$1 per unit time while in service, yields
 average number in service = $\lambda(1 - P_N)E[S]$.

Also, if we suppose that each customer in the system pays $$x$ per unit time whenever its remaining service time is x , then we get

$$V = \lambda(1 - P_N)E\left[SW_Q^* + \int_0^S (S - x)dx\right] = \lambda(1 - P_N)[E[S]W_Q + E[S^2]/2]$$

where V is the (time) average amount of work in the system and where W_Q^* is a random variable representing the (limiting) amount of time that the n th entering customer spends waiting in queue.

The above gives us one equation relating V and W_Q^* and one approach to obtaining W_Q^* would be to derive a second equation. An approximate second equation was given by Nozaki-Ross in [3] by means of the following approximation assumption.

Approximation Assumption

Given that a customer arrives to find i busy servers, $i > 0$, at the time he enters service the remaining service times of the other customers being served are approximately independent each having the equilibrium service distribution.

Using the above assumption as if it was exactly true, Nozaki-Ross were able to derive a second relationship between V and W_Q^* which resulted in an expression for W_Q^* as a function of P_N . By approximating P_N by its known value in the case where the service distribution is exponential, Nozaki-Ross came up with the following approximation for W_Q^* .

$$W_Q^* = \frac{\frac{E[S]^2}{2E[S]} \sum_{j=k}^{N-1} \frac{(\lambda E[S])^j}{k! k^{j-k}} - (N - k) \frac{E[S](\lambda E[S])^N}{k! k^{N-k}}}{\sum_{j=0}^{k-1} \frac{(\lambda E[S])^j}{j!} + \sum_{j=k}^{N-1} \frac{(\lambda E[S])^j}{k! k^{j-k}} (k - \lambda E[S])}.$$

The idea of an approximation assumption to approximate various quantities of interest of the model M/G/k was also used by Tijms, Van Hoorn, and Federgruen [6]. They used a slightly different approximation assumption to obtain approximations for the steady state probabilities. Other approximations for the M/G/k are also given in Boxma, Cohen, and Huffels [1], and Takahashi [5].

References

- [1] Boxma, O. J., Cohen, J. W., and Huffels, N.: 1979,
Operations Research 27, pp. 1115-1127.
- [2] Kelley, F.: 1979, REVERSIBILITY AND STOCHASTIC NETWORKS,
John Wiley & Sons.
- [3] Nozaki, S. A., and Ross, S. M.: 1978, Journal of Applied
Probability 15, pp. 826-834.
- [4] Ross, S. M.: 1980, INTRODUCTION TO PROBABILITY MODELS,
Second Edition, Academic Press.
- [5] Takahashi, Y.: 1977, Journal of the Operations Research
Society of Japan 20.
- [6] Tijms, H. C., Van Hoorn, M. H., and Federgruen, A.: 1979,
"Approximations for the Steady State Probabilities in
the M/G/c queue," Research Report 49, der Vrije
University, to appear Advances in Applied Probability.

